

Vorwort

Hallo liebe Tutorial Freunde,

heute werde ich euch nach langer Pause in Sachen Anleitungen den wohl nützlichsten Bereich der auf Ribon, Mangos und Trinity basierenden DBs erklären mit dem ihr ganz ohne C++ Scripts und nur mit der DB richtige Gossip NPCs erstellen könnt.

Ich werde wie immer in diesem Tutorial sämtliche Inhalte des Gossip Systems erklären und anhand von Beispielen weiter ausführen.

[How to] Gossip Sytsem



Bevor wir ein Script anfangen sollten wir uns zuerst einmal überlegen was unser Script und auch NPC der das ganze machen soll, für Einstellungen brauchen.

Ich möchte heute einen Allround NPC machen der Teleportieren, Spellen, Verkaufen und Quest ausgeben kann. Dabei muss ich beachten das ich im Gossip Script und auch bei den NPC Flags diese Punkte auch einhalte damit er am ende auch richtig funktioniert.

Fazit bei meinem Vorhaben:

NPC braucht die Flags:

- (1) Gossip
- (2) Questgiver
- (128) Vendor
- (4096) Repair

Gossip Menü braucht:

Menü für Questausgabe
Menü für Vendor
Menü für Teleports
Menü für Zauber

Soweit wissen wir jetzt was wir brauchen, nun kommen wir zur Umsetzung des ganzen Systems. Wir erstellen uns also erst mal unseren NPC:

[illegible]

Gossip Menü ID | Npc Flags

Wichtig ist das ihr die Flags richtig bestimmt und der Npc eine feste Gossip Menü ID bekommt, der Rest bleibt euch selbst überlassen.

Meine Gossip ID die ich gewählt habe ist 6198 mit ihr werde ich nun das Gossip Menü aufbauen.

Dazu brauchen wir ein Hauptmenü das wir unter der Tabelle `gossip_menu` erstellen. Zusätzlich möchte ich das mein NPC in Hauptmenü etwas sagt wofür ich als alles erstes einmal einen Text erstellen werde.

Beschreibung Tabelle npc_text:

ID:

Die ID des Textes, merkt sie euch, den ihr braucht sie später noch einmal im Hauptmenü.

text0_0:

Hier tragt ihr den Text ein den euer NPC sagen soll.

text0_1:

Den selben Text nochmal hier eintragen!

prob0:

Die Wahrscheinlichkeit mit der der NPC den Text sagen wird.
In unserem fall immer eine 1.

em0_0:

Steuert ob der Text bei der Begrüßung angezeigt werden soll.
In dem fall auch **1** für ja.

Nun sind wir mir der Erstellung des Begrüßungstextes fertig, mein Beispiel sieht wie folgt aus:

[illegible]

Speichert das ganze schon mal ab !

Weiter geht es mit der Erstellung des eigentlichen Hauptmenüs.

Beschreibung Tabelle gossip_menu:

entry:

Die Gossip ID die ihr euch ausgesucht habt und zuvor eurem NPC zugewiesen habt, wird hier eingetragen.

Es kann mehrere Entrys mit derselben Gossip ID in allen gossip Tabellen geben !!!

text id:

Hier müsst ihr die ID eures zuvor erstellten Textes der **npc_text** Tabelle angeben.

Nun haben wir schon den Grundstein und können darauf nun ein Gerüst aufbauen.

Hier wieder mein Beispiel:

```
DELETE FROM `gossip_menu` WHERE (`entry`=6198);  
INSERT INTO `gossip_menu` VALUES ('6198', '52000', '0', '0', '0', '0', '0', '0');
```

**Man kann es sich wie einen Stammbaum vorstellen auf dem das ganze aufgebaut wird.
Nun machen wir weiter mit dem Hauptmenü Verzeichnis, dazu gehen wir in die Tabelle [gossip_menu_option](#).**

Beschreibung Tabelle gossip_menu_option:

menu id:

Ist wieder eure Gossip ID vom NPC.

id:

Ist die Position des Verzeichnis Punktes.

Wen ein Menü mehr Punkte hat müsst ihr streng darauf achten das ihr diese auch immer ansteigen lasst, also erster Punkt 1 zweiter Punkt 2 usw.

option icon:

Das Symbol des Verzeichnis Punktes.

Es gibt folgende Symbole:

GOSSIP_ICON_CHAT	= 0,	//white chat bubble
GOSSIP_ICON_VENDOR	= 1,	//brown bag
GOSSIP_ICON_TAXI	= 2,	//flight
GOSSIP_ICON_TRAINER	= 3,	//book
GOSSIP_ICON_INTERACT_1	= 4,	//interaction wheel
GOSSIP_ICON_INTERACT_2	= 5,	//interaction wheel
GOSSIP_ICON_MONEY_BAG	= 6,	//brown bag with yellow dot
GOSSIP_ICON_TALK	= 7,	//white chat bubble with black dots
GOSSIP_ICON_TABARD	= 8,	//tabard
GOSSIP_ICON_BATTLE	= 9,	//two swords
GOSSIP_ICON_DOT	= 10,	//yellow dot

option text:

Der Name des Punktes.

option id:

Die Option ID der Gossip ID.

Mit ihr legt ihr fest welche Option dieser Punkt hat,

Bei einem Normalem Gossip wäre hier die 1 nötig aber wen der NPC z.B. auch Quest einteigen soll brauchen wir auch einen Punkt mit der Option 2 für Questgeber.

```

GOSSIP_OPTION_NONE           = 0,    //UNIT_NPC_FLAG_NONE           (0)
GOSSIP_OPTION_GOSSIP         = 1,    //UNIT_NPC_FLAG_GOSSIP         (1)
GOSSIP_OPTION_QUESTGIVER     = 2,    //UNIT_NPC_FLAG_QUESTGIVER     (2)
GOSSIP_OPTION_VENDOR         = 3,    //UNIT_NPC_FLAG_VENDOR         (128)
GOSSIP_OPTION_TAXIVENDOR     = 4,    //UNIT_NPC_FLAG_TAXIVENDOR     (8192)
GOSSIP_OPTION_TRAINER        = 5,    //UNIT_NPC_FLAG_TRAINER        (16)
GOSSIP_OPTION_SPIRITHEALER    = 6,    //UNIT_NPC_FLAG_SPIRITHEALER    (16384)
GOSSIP_OPTION_SPIRITGUIDE     = 7,    //UNIT_NPC_FLAG_SPIRITGUIDE     (32768)
GOSSIP_OPTION_INNKEEPER      = 8,    //UNIT_NPC_FLAG_INNKEEPER      (65536)
GOSSIP_OPTION_BANKER         = 9,    //UNIT_NPC_FLAG_BANKER         (131072)
GOSSIP_OPTION_PETITIONER     = 10,   //UNIT_NPC_FLAG_PETITIONER     (262144)
GOSSIP_OPTION_TABARDDESIGNER = 11,   //UNIT_NPC_FLAG_TABARDDESIGNER (524288)
GOSSIP_OPTION_BATTLEFIELD    = 12,   //UNIT_NPC_FLAG_BATTLEFIELDPERSON (1048576)
GOSSIP_OPTION_AUCTIONEER     = 13,   //UNIT_NPC_FLAG_AUCTIONEER     (2097152)
GOSSIP_OPTION_STABLEPET      = 14,   //UNIT_NPC_FLAG_STABLE         (4194304)
GOSSIP_OPTION_ARMORER        = 15,   //UNIT_NPC_FLAG_ARMORER        (4096)
GOSSIP_OPTION_UNLEARNTALENTS = 16,   //UNIT_NPC_FLAG_TRAINER        (16) (bonus option for GOSSIP_OPTION_TRAINER)
GOSSIP_OPTION_UNLEARNPETTALENTS = 17, //UNIT_NPC_FLAG_TRAINER        (16) (bonus option for GOSSIP_OPTION_TRAINER)
GOSSIP_OPTION_LEARNDUALSPEC  = 18,   //UNIT_NPC_FLAG_TRAINER        (16) (bonus option for GOSSIP_OPTION_TRAINER)
GOSSIP_OPTION_OUTDOORPVP     = 19,   //added by code (option for outdoor pvp creatures)
GOSSIP_OPTION_MAX

```

npc option npcflag:

Auch hier ist es wichtig das ihr die Flag zum Punkt setzt.

Dieser Punkt regelt welche Flag der NPC benutzen soll wen man diesen Menüpunkt anklickt. Soll der NPC also auch Quest ausgeben können braucht er die NPCflags für Quest (2) die wir hier dann ebenso eintragen müssen, Hier dürft ihr immer nur eine Flag benutzen, hier niemals Flags Addieren !!!

action menu id:

Hier wird die Gossip ID für das nachfolgende Menü eines Punktes angegeben das angezeigt werden soll wen man auf diesen Punkt klickt.

Manche Punkte wie z.B. für ein Quest Gossip wird dieses Feld nicht benötigt das es von da ab direkt in den Questlog geht, dieses Feld ist nur für Punkte mit Untermenüs gut wie z.B. Teleports.

action poi id:

Interessant für Leute die gerne ihren eigenen Wegweiser machen wollten.

Hier kann man die ID eines Eintrages aus der points_of_interest Tabelle angeben der Dann auf der Minimap angezeigt wird wen man auf diesen Menüpunkt klickt.

Die meisten Wachen wie die in Dalaran funktionieren schon so, wer sich das ganze also mal genauer ansehen will schaut sich deren Gossip Menu an.

action script id:

Hier wird die ID des Scripts aus der gossip_script Tabelle angegeben.

Sie ist nützlich für Gossip Scripts mit Event wie z.B. einem Port, Movement uvm.

Teleporter und Spell Gossip Scripts brauchen hier eine ID und das nötige Script.

Dazu aber später.

box coded:

Entweder 1 für aktiv oder 0 für deaktiviert..

Dieses Feld erlaubt es euch für den Menüpunkt eine Box auszugeben wie z.B. die Box bei der Abfrage zum Kauf von Dualspec.

box text:

Hier könnt ihr einen Text für eine Box eintragen die beim aktivieren dieses Menüpunktes ausgegeben werden kann. Ich habe so in meiner DB ein fast 100% Blizzlike Menü gebaut für den Kauf von Dualspec.

box money:



Das Geld das in der Box angezeigt wird und das man braucht um den Menüpunkt Fortzusetzen.

Hier seht ihr z.B. mein Dualspec Menü das ich mit der Box bei mir in der DB habe.
100% Blizzlike Style.

cond 1 bis cond 3:

Eine weitere sehr interessante Funktion der Menüpunkte sind die Conditions.

Heißt so viel wie Voraussetzungen die man haben muss um den Menüpunkt überhaupt zu sehen oder zu aktivieren.

CONDITION_NONE	= 0,	// 0	0	0	always true
CONDITION_AURA	= 1,	// spell_id	effindex	0	true if has aura of spell_id with effect effindex
CONDITION_ITEM	= 2,	// item_id	count	0	true if has count of item_ids
CONDITION_ITEM_EQUIPPED	= 3,	// item_id	0	0	true if has item_id equipped
CONDITION_ZONEID	= 4,	// zone_id	0	0	true if in zone_id
CONDITION_REPUTATION_RANK	= 5,	// faction_id	min_rank	0	true if has min_rank for faction_id
CONDITION_TEAM	= 6,	// player_team	0,	0	469 - Alliance, 67 - Horde)
CONDITION_SKILL	= 7,	// skill_id	skill_value	0	true if has skill_value for skill_id
CONDITION_QUESTREWARDED	= 8,	// quest_id	0	0	true if quest_id was rewarded before
CONDITION_QUESTTAKEN	= 9,	// quest_id	0,	0	true while quest active
CONDITION_AD_COMMISSION_AURA	= 10,	// 0	0,	0	true while one from AD commission aura active
CONDITION_NO_AURA	= 11,	// spell_id	effindex	0	true if does not have aura of spell_id with effect effindex
CONDITION_ACTIVE_EVENT	= 12,	// event_id	0	0	true if event is active
CONDITION_INSTANCE_DATA	= 13,	// entry	data	0	true if data is set in current instance
CONDITION_QUEST_NONE	= 14,	// quest_id	0	0	true if doesn't have quest saved
CONDITION_CLASS	= 15,	// class	0	0	true if player's class is equal to class
CONDITION_RACE	= 16,	// race	0	0	true if player's race is equal to race
CONDITION_ACHIEVEMENT	= 17	// achievement_id	0	0	true if achievement is complete

cond 1 val 1 usw:

Die letzten Felder beinhalten immer die nötigen Informationen für die in cond_ Angegebene Voraussetzung. Habt ihr z.B. 2 (CONDITION_ITEM) bei Cond_1 eingetragen so muss bei cond_val_1 die Item Entry ID rein und bei cond_val_2 die Mengenangabe wie viel man von diesem Item haben muss.

So viel zur Tabelle gossip_menu_option, nun gehen wir an mein Bsp. Und schauen uns mal sein Hauptmenü an:

```
-- Hauptmenu Verzeichniss
DELETE FROM `gossip_menu_option` WHERE `menu_id` = 6198;
INSERT INTO `gossip_menu_option` VALUES ('6198', '0', '0', 'GOSSIP_OPTION_QUESTGIVER', '2', '2', '0', '0', '0', '0', '0', '0', null, '0', '0', '0', '0', '0', '0', '0', '0');
INSERT INTO `gossip_menu_option` VALUES ('6198', '1', '2', 'Teleports', '1', '1', '6199', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0');
INSERT INTO `gossip_menu_option` VALUES ('6198', '2', '3', 'Zauber', '1', '1', '6203', '0', '0', '0', '0', '0', null, '0', '0', '0', '0', '0', '0', '0', '0');
INSERT INTO `gossip_menu_option` VALUES ('6198', '3', '1', 'Ich möchte mein Start Equip', '3', '4224', '0', '0', '0', '0', '0', '0', null, '0', '0', '0', '0', '0', '0', '0', '0');
```

menu_id | id | option_icon | option_id | option_npcflag | action_menu_id

Wie ihr sehen könnt haben diese Punkte nun eine action_script_id die den Punkten ein Script zuweisen, diese Scripts werden wir nun anlegen:
Dazu gehen wir in die Tabelle **gossip_scripts**.

Beschreibung Tabelle gossip_scripts:

id:

ID des Gossip Scripts die ihr euch vorher wieder ausdenken musstet.

delay:

Zeit die vergeht bis das Script nach der Aktivierung ausgeführt wird in Sekunden.

command:

Hier kommt die Nummer des Befehls rein den dieses Script ausführen soll:

0	TALK	Creature say/whisper/yell/textemote.
1	EMOTE	Play emote on creature.
2	FIELD_SET	Change the value at an index for the player.
3	MOVE_TO	Relocate creature to a destination
4	FLAG_SET	Turns on bits on a flag field at an index for the player..
5	FLAG_REMOVE	Turns off bits on a flag field at an index for the player.
6	TELEPORT_TO	Teleports the player to a location.
7	QUEST_EXPLORFD	Satisfies the explore requirement for a quest
9	RESPAWN_GAMEOBJECT	Spawns a despawned gameobject.
10	TEMP_SUMMON_CREATURE	Temporarily summon a creature.
11	OPEN_DOOR	Opens a door gameobject (type == 0).
12	CLOSE_DOOR	Closes a door gameobject (type == 0).
13	ACTIVATE_OBJECT	Activates an object.
14	REMOVE_AURA	Removes an aura due to a spell.
15	CAST_SPELL	Casts a spell.
16	PLAY_SOUND	Plays a sound
17	nuttin	There is no command 17
18	nuttin	There is no command 18
19	nuttin	There is no command 19
20	LOAD_PATH	Load path to unit, then unit start waypoint movement.
21	CALLSCRIPT_TO_UNIT	No idea what this does right now -- sure sounds cool though.
22	KILL	Kills something :)

datalong, datalong2 und X,Y,Z & O::

Je nach dem welchen Command ihr ausführen lassen wollt muss hier entsprechender Wert der von dem Command verlangt wird eingetragen werden.

Depending on what command was used, the meaning and use for the following fields varies.

• SCRIPT_COMMAND_TALK = 0

- datalong: 0=say, 1=whisper, 2=yell, 3=text emote
- dataint: The text ID that the creature will say. See [db_script_string](#). ID must be between 2000000000 and 2000010000.

• SCRIPT_COMMAND_EMOTE = 1

- datalong: The emote ID to play.

• SCRIPT_COMMAND_FIELD_SET = 2

- datalong: Index of the field.
- datalong2: Value to place at the index.

• SCRIPT_COMMAND_MOVE_TO = 3

- datalong2: Length of the motion.
- x: X position to move to.
- y: Y position to move to.
- z: Z position to move to.
- o: Orientation to face.

• SCRIPT_COMMAND_FLAG_SET = 4	<ul style="list-style-type: none"> o datalong: Field index to be set. o datalong2: Flag bit(s) to set.
• SCRIPT_COMMAND_FLAG_REMOVE = 5	<ul style="list-style-type: none"> o datalong: Field index to be unset. o datalong2: Flag bit(s) to unset.
• SCRIPT_COMMAND_TELEPORT_TO = 6	<ul style="list-style-type: none"> o datalong: Target Map ID. See Maps.dbc o x: Teleport target x coordinate. o y: Teleport target y coordinate. o z: Teleport target z coordinate. o o: Teleport target orientation.
• SCRIPT_COMMAND_QUEST_EXPLORED = 7	<ul style="list-style-type: none"> o datalong: Quest ID whose external status should be satisfied o datalong2: Distance away from the NPC/object that the player can be and have the script still take effect
• SCRIPT_COMMAND_KILL_CREDIT = 8	<ul style="list-style-type: none"> o datalong: Kill credit entry for quest (entry in quest_template.ReqCreatureOrGOId) o datalong2: 0=personal credit, 1=group credit
• SCRIPT_COMMAND_RESPAWN_GAMEOBJECT = 9	<ul style="list-style-type: none"> o datalong: Guid of the gameobject to respawn. See gameobject.guid. o datalong2: Despawn time in seconds. If the value is < 5 seconds: 5 is used instead.
• SCRIPT_COMMAND_TEMP_SUMMON_CREATURE = 10	<ul style="list-style-type: none"> o datalong: Entry of the summoned creature from creature_template.entry. o datalong2: Despawn time in ms. o x: Summon target x coordinate. o y: Summon target y coordinate. o z: Summon target z coordinate. o o: Summon target orientation.
• SCRIPT_COMMAND_OPEN_DOOR = 11	<ul style="list-style-type: none"> o datalong: Guid of the activated door. It's a gameobject.guid. o datalong2: Delay before closing again the door. If the value is < 5 seconds: 5 is used instead.
• SCRIPT_COMMAND_CLOSE_DOOR = 12	<ul style="list-style-type: none"> o datalong: Guid of the activated door. It's a gameobject.guid. o datalong2: Delay before opening again the door. If the value is < 5 seconds: 5 is used instead.
• SCRIPT_COMMAND_ACTIVATE_OBJECT = 13	<ul style="list-style-type: none"> o The source must be a unit and the target a game object.
• SCRIPT_COMMAND_REMOVE_AURA = 14	<ul style="list-style-type: none"> o datalong: Spell ID. o datalong2: If value > 0, then cast on the source; otherwise cast on the target.
• SCRIPT_COMMAND_CAST_SPELL = 15	<ul style="list-style-type: none"> o datalong: Spell ID. o datalong2: If value > 0, then cast on the source; otherwise cast on the target.

- **SCRIPT_COMMAND_PLAY_SOUND = 16**

- datalong: Sound ID.
- datalong2: bitmask: 0/1=anyone/target, 0/2=with distance dependent, so 1|2 = 3 is target with distance dependent.

Side note: Source may be any object/npc. Target may be any player.

- **SCRIPT_COMMAND_CREATE_ITEM = 17**

- datalong: item entry.
- datalong2: amount.

Side note: Source or target have to be a player.

- **SCRIPT_COMMAND_DESPAWN_SELF = 18**

- datalong: despawn delay.

Side note: Source or target must be creature.

Bei mir sehen die fertigen Teleport Scripts nun so aus:

```
-- Teleport Shattrath
DELETE FROM `gossip_scripts` WHERE `id` = 50022;
INSERT INTO `gossip_scripts` VALUES ('50022', '0', '6', '530', '0', '0', '-1887.510010', '5359.379883', '-12.427300', '0.000000');

-- Teleport Dalaran
DELETE FROM `gossip_scripts` WHERE `id` = 50023;
INSERT INTO `gossip_scripts` VALUES ('50023', '0', '6', '571', '0', '0', '5804.182129', '624.308594', '647.786865', '1.640000');
```

Nun habt ihr den Teleport Teil unseres Gossip Npcs fertig und könnt ihn wenn ihr nichts falsch gemacht habt benutzen.

Kommen wir zum letzten Teil, den Spells.

Es ist nicht viel anders als bei den Teleports, es muss nur ein anderer Command verwendet werden, in dem Fall brauchen wir bei allen gossip_scripts den Command 14 für COMMAND_CAST_SPELL und falls wir noch die Wiederbelebungs-nachwirkungen entfernen wollen den Command 15 für COMMAND_REMOVE_AURA.

Wir bauen also erst mal wie gewohnt das Menü für die Zauber, bei meinem Bps. Hate es die Gossip Menü ID 6203:

```
-- Zauber
DELETE FROM `npc_text` WHERE `ID` = 52002;
DELETE FROM `gossip_menu` WHERE (`entry`=6203);
INSERT INTO `npc_text` VALUES ('52002', 'Welcher Zauber darfs den sein ?', 'Welcher Zauber darfs den sein ?',
'0', '1', '0', '1', '0', '0', '0', '0', ',', ',', '0', '0', '0', '0', '0', '0', '0', '0', ',', ',', '0', '0', '0', '0', '0', '0', '0', '0', ',', ',', '0', '0', '0', '0',
'0', '0', '0', '0', ',', ',', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0',
',', ',', '0', '0', '0', '0', '0', '0', '0', '0');
INSERT INTO `gossip_menu` VALUES ('6203', '52002', '0', '0', '0', '0', '0', '0');
```

```
-- Zauber Verzeichniss
DELETE FROM `gossip_menu_option` WHERE `menu_id` = 6203;
INSERT INTO `gossip_menu_option` VALUES ('6203', '7', '7', 'Hauptmenü', '1', '1', '6198', '0', '0', '0', '0', '', '0', '0', '0', '0', '0', '0', '0', '0');
INSERT INTO `gossip_menu_option` VALUES ('6203', '2', '3', 'Wiederbelebungs nachwirkungen entfernen', '1', '1', '0', '0', '50029', '0', '0', '', '1', '15007', '0', '0', '0', '0', '0', '0', '0');
INSERT INTO `gossip_menu_option` VALUES ('6203', '3', '3', 'Großer Segen der Könige', '1', '1', '0', '0', '50030', '0', '0', '', '0', '0', '0', '0', '0', '0', '0', '0');
```

Achtet wieder auf das bisher gelehnte um nichts falsch zu machen !!!

Wie ihr seht hat der Menüpunkt Wiederbelebungs nachwirkungen entfernen angegeben bei den Conditions, ich habe hier eingestellt das man den Menüpunkt nur sieht wen man den Debuff auch wirklich hat.

Nun zu den eigentlichen Zauber Scripts die wir wieder in der gossip_scripts Tabelle machen:

```
-- Zauber Wiederbelebungs nachwirkungen entfernen
DELETE FROM `gossip_scripts` WHERE `id` = 50029;
INSERT INTO `gossip_scripts` VALUES ('50029', '0', '14', '15007', '0', '0', '0', '0', '0', '0');

-- Zauber Großer Segen der Könige
DELETE FROM `gossip_scripts` WHERE `id` = 50030;
INSERT INTO `gossip_scripts` VALUES ('50030', '0', '15', '25898', '0', '0', '0', '0', '0', '0');
```

Unter Datalong must ihr diesmal nur die Spell ID eingeben und datalong2 bleibt bei 0, stellt ihr datalong2 beim Command 15 höher als auf 0 so wird der Spell nicht auf den gewirkt den denn Menüpunkt ausführt sondern auf den NPC selbst.

Lange rede und viele Zahlen bringen mich nun schließlich zum Ende meines kleinen Tutorials.





Ich bedanke mich für eure Aufmerksamkeit und hoffe das ich vielen damit helfen kann sich mal etwas mehr mit der DB zu spielen. Viel Erfolg beim erstellen eurer Scripts

MFG

euer Steve 🍷 🍷 🍷